Template of Hardware Metapaper

For submission to the *Journal of Open Hardware*

To complete this template, please replace the blue text with your own. The paper has five main sections: (1) Overview; (2) Quality Control; (3) Application (4) Build Details (5) Discussion.

# (1) Overview

## Title

The title of the hardware paper should focus on the hardware, e.g. “[type of hardware] to automate laboratory protocol X” or “Surface Trawl for Collecting Marine Microplastics.” If the hardware is closely linked to a specific research paper, then “[Hardware type] from [Paper Title]” is appropriate. The title should relate to the functionality of the hardware and the area it relates, to rather than making claims about the usability or user characteristics of hardware, e.g. “Easy-to-use”.

## Sport Analyzer: Multi-sensor device for data acquisition and visualization in sport performance and rehabilitation.

## Paper authors

1. Last name, first name; *(Lead/corresponding author first)*

2. Last name, first name; *etc.*

1. Cano-Ferrer Xavier (First/corresponding Author)

2. Imbert, Albane

## Paper author roles and affiliations

1. First author role and affiliation

2. Second author role and affiliation *etc.*

1. Research Engineer, Making Science Technology Platform, The Francis Crick Institute, London, United Kingdom

2. Head, Making Science Technology Platform, The Francis Crick Institute, London, United Kingdom

*Note: One author can have multiple affiliations and affiliations can include “independent”.*

## Abstract

A short (up to 250 words) summary of the hardware being described: what problem(s) the hardware addresses, what it does, how it technically/methodologically advances the state-of-the-art, how it was designed and implemented, and its applicability to other issues/research/areas of reuse.

In this paper we introduce Sport Analyzer, a battery powered device which records and displays force and speed signals simultaneously and a small footprint. It provides a universal interface circuit for optical incremental encoders and load cells and calibration guides that will allow sport scientists, physiotherapists and medical scientists to build their own hardware and understand how to convert an industrial sensor in a human movement device.

|  |
| --- |
| Metadata Overview *Please provide the following overview and self-assessment*  Main design files: link to repository with design files and assembly instructions <https://github.com/XaviCanoFerrer/Sport_Analyzer>  Target group: e.g. secondary school students, layperson, undergraduate students, scientists in [discipline], trained engineers, use of professional services  Scientists, professionals or undergraduate students in the following disciplines: sport science, sports medicine, physiotherapy, biomechanics.  Skills required: For each main manufacturing method, provide information pairs (method - easy / advanced / specialist). E.g. desktop 3d printing - easy; surface mount PCB - advanced; injection moulding – specialist  Laser ablation – easy; Mechanical assembly – easy; Electrical assembly – advanced; Software – easy.  Replication: Project has been replicated at time of publication? Include reference (possibly way of contact). Links to places where future builds and kits might be found (repository, group website, collaboration site, etc.).  *See section “Build Details” for more detail.*  No builds known to the authors so far. Device has been used by authors for a paper currently in preparation. Keywords *(required)* keyword 1; keyword 2; etc.  Keywords should make it easy to identify who and what the hardware will be useful for. Maximum of 5 keywords. E.g.: microbiology; laboratory; microfluidics; single cell analysis; fluorescence  Biomechanics, power-oriented resistance training, rehabilitation, sport science, encoder. |

## 

## Introduction

An overview of the hardware, what it does, how it was produced, and the research for which it has been used. If the hardware addresses a specific lack or problem, describe the problem in this section and how the hardware addresses the problem. Elaborate how it technically/methodologically advances the state-of-the-art including references to relevant research articles and online references.

Between all sensors used for sport performance and rehabilitation, incremental encoders and load cells are some of the most effective to measure velocities and forces associated with physical evaluation tests or exercises. In the case of linear and rotary encoders, they have been extensively used in sport sciences publications to predict the maximal dynamic strength of athletes from the Load-Velocity relationship [1, 2, 3], to evaluate reliability of other velocity and power measuring devices [4, 5] and measure concentric and eccentric power in inertial squat performance [6, 7] between other applications. Load cells have also been used extensively in multiple applications such as: measurement of maximum voluntary contraction or isometric strength (MVC) [8], the instantaneous and average force applied during concentric and eccentric contractions in inertial exercises [6] or simply measure forces from different nature such in the case of elastic bands [9]. The aim of the present work is to describe how to manufacture and reproduce a hardware-based device which can display and record signals from both sensors.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Sport Analyzer | Chronojump | Musclelab | Vitrube | Gymaware |
| Load cell | Y | Y | Y | N | N |
| Linear encoder | Y | Y | Y | Y | Y |
| Rotary encoder | Y | Y | Y | N | N |
| Battery powered | Y | N | N | Y | Y |
| Requires computer software | N | Y | Y | Y | Y |
| Open source software | Y | Y | N | N | N |
| Open source hardware | Y | N | N | N | N |
| GUI for data visualization | N | Y | Y | Y | Y |
| Cost (GBP) | 179.99 (parts) | 773.22 (from Chronojump) | 2,467.77 (from Simplifaster) | 339.75 (from Virtube) | 2,353.43 (from Gymaware) |

Problematic: Systems available are expensive or computer dependent and no hardware based

***Table of the different commercial options***

***Figure with examples of figures of exercises to show applications***

The relationship between power output and sport specificity was introduced by Bosco, Luhtanen and Komi in 1983 Velocity-Based Training

***Table of capabilities***

## Overall Implementation and Design

Describe how the hardware was implemented/created, with relevant details of the architecture and design, including general materials (*Note: specific details on materials can be addressed in later section “Availability of materials and methods”*). The use of diagrams and pictures of the assembled hardware is appropriate. Please also describe any variants and associated implementation differences.

Sport Analyzer is a programmable, Arduino compatible and battery powered device for simultaneous data collection and visualization of two important variables in sport sciences and rehabilitation which are force and velocity. The device accomplishes it by combining a load cell and optical incremental encoder data acquisition (Figure 1a).

System Architecture

The device has been designed to provide the visual image of a linear or rotary encoder and a load cell signals and at the same time acquire their data and save it inside a MicroSD card. It provides the simplicity of having one pushbutton to change the signal displayed and another to place a numeric label associated with the number of sample acquired. The device is battery operated using a regular 9V alkaline battery which also provides availability (Figure 1b).

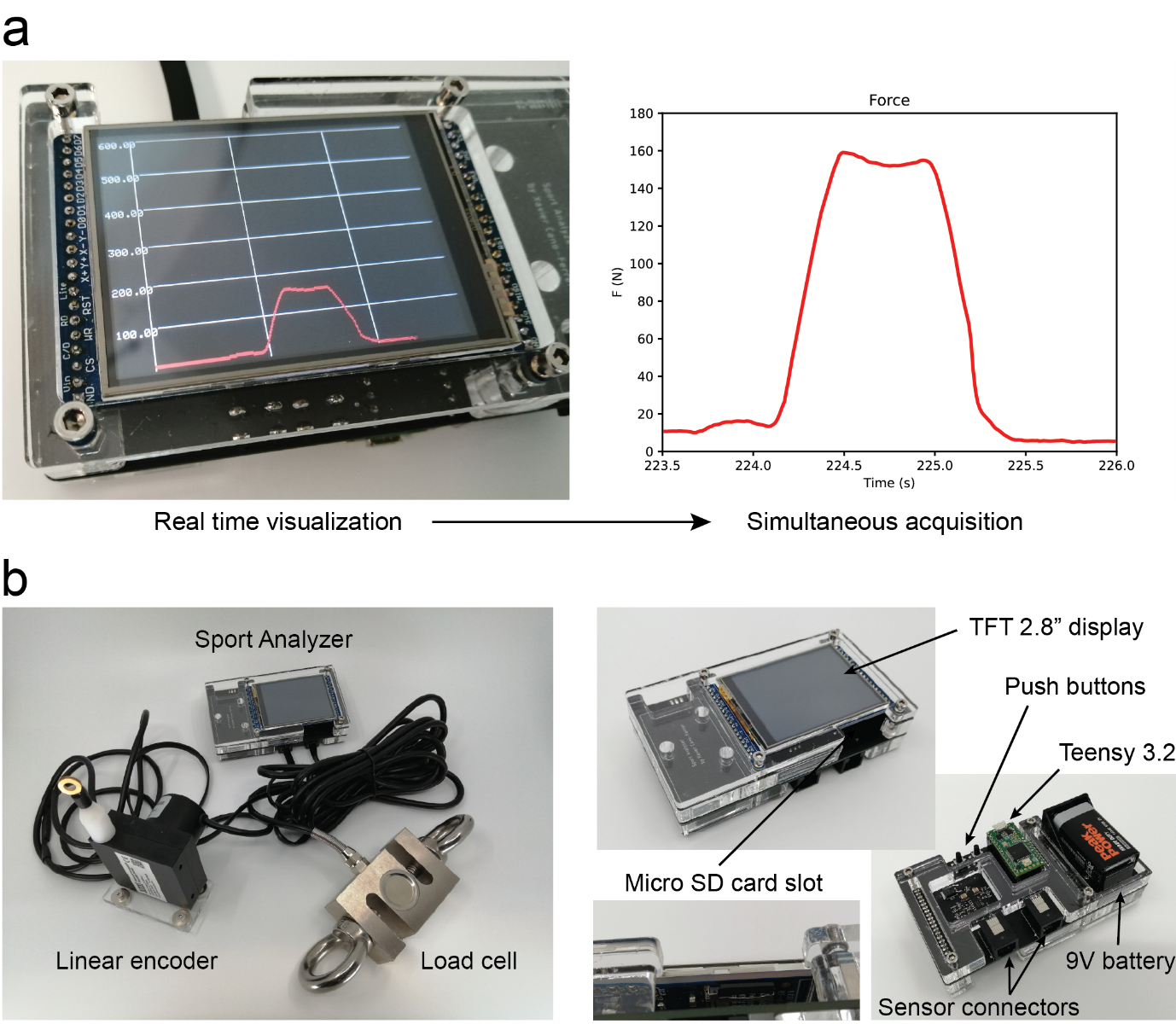


Figure 1: a. Sport Analyzer allows simultaneous data visualization and acquisition for its further analysis. The figure shows the same signal during the acquisition on the TFT display (left) and after the analysis in a plot from the data extracted from the txt. file (right). b. The device with the linear encoder and the load cell connected (left) and the main components of the device (right).

Electronics

The device main control unit is the Teensy 3.2 development board (Freescale Semiconductor MK20DX256). It has two RJ12 connectors which allow the connections for the Linear or rotary encoder and the load cell. It has a TFT 2.8” display (Adafruit 2.8" TFT LCD with Touchscreen Breakout Board with MicroSD Socket - ILI9341) which also has a MicroSD card socket where the device stores the data collected from the sensors in a txt. file. The device has two pushbuttons, the first one allows the user to switch from displaying the encoder signal (in color blue) and the load cell signal (in color red) on the TFT screen. The second pushbutton increments a counter to label the first column of data of the txt. file stored on the MicroSD card. The load cell conditioning signal is operated by a HX711 (24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales, Avia Semiconductor). The optical incremental encoder signal admits 5V A, B and Z phases connection directly on the Teensy 3.2 pins, which are 5V tolerant. The device is powered using a regular 9V battery (Figure 2). The optical incremental encoder used in the present work is a Wisamic with 2400 pulses per revolution with a response frequency of (0-20) kHz and a maximum speed of 6000 rpm. On the other hand, the load cell is an S-type load cell which allows compression and tension forces measurement. The load cell main electrical properties are defined by the sensitivity of 2.0mV/V, repeatability of ±0.02(%F.S.), Input impedance of 350±10Ω, Linearity of ±0.02(%F.S.) and a measuring range of 4.9-4900 N.

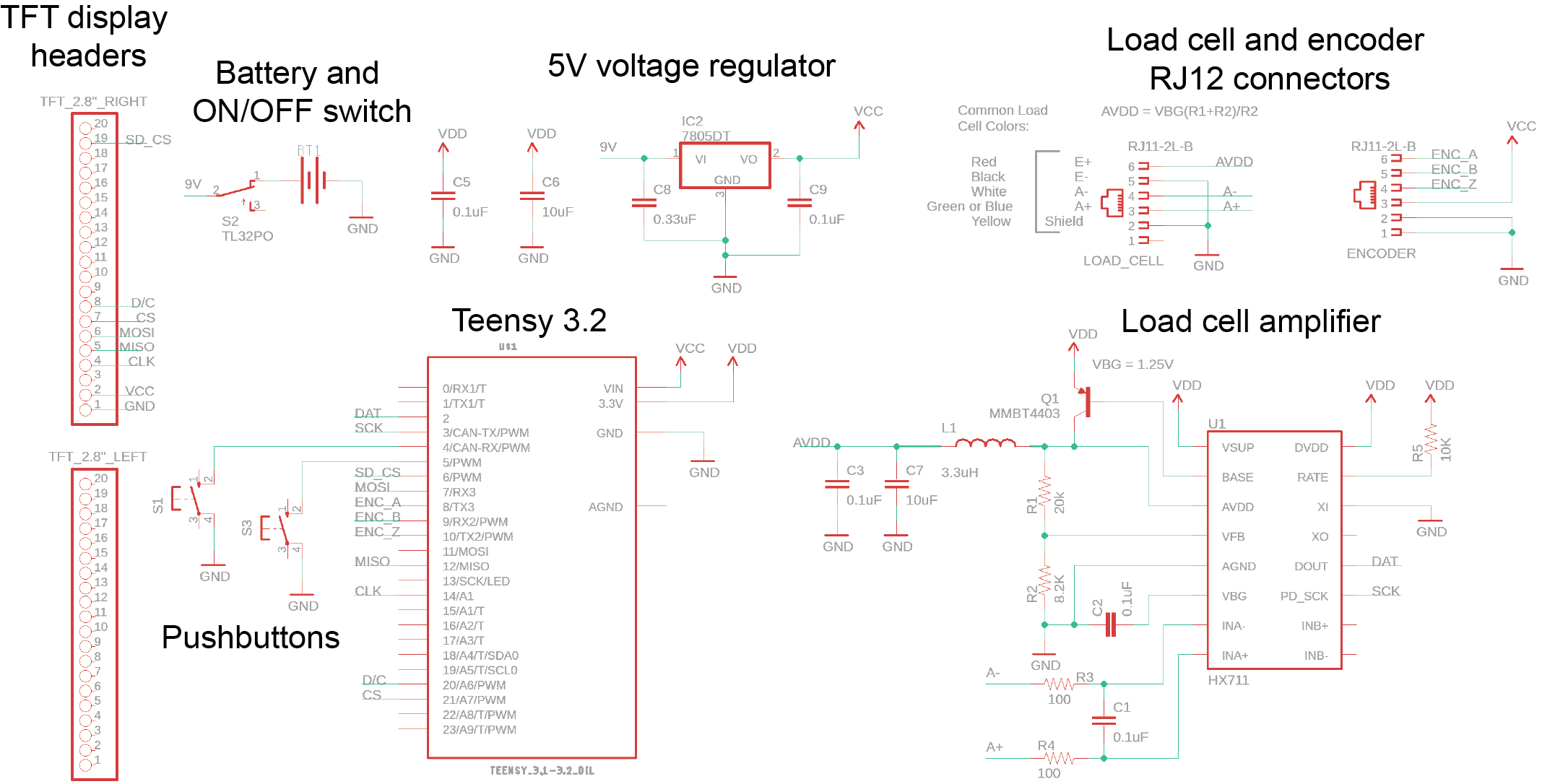


Figure 2: Electronic schematics of the device. Sport Analyzer schematic is composed by the Load cell amplifier circuit, the Teensy 3.2 development board, two pushbuttons, an ON/OFF switch, the two RJ12 sensor connectors, a 5V voltage regulator which converts the 9V of the battery and the TFT display pin headers.

Firmware

The firmware which is constantly running on the main loop function on the Teensy 3.2 is composed by its main file: Sport\_Analayzer\_1.0.ino which uses mainly six libraries: the SPI library [10], the ILI9341\_t3 library [11], the font\_Arial library [12], the Bounce library [13], the Encoder library [14], the SD library [15] and the HX711 library [16]. A part from the libraries, there are essential parts of the firmware that are based on the work of other authors, this functions are: the function that generates a graph for TFT displays [17] and the application we use to generate the code from the Sport Analyzer logo bitmap [18].

The main loop function of the firmware runs inside a three hundred samples for loop which is enough to fill the horizontal axis of the screen. For every iteration, a reading of the pushbuttons, load cell and encoder is performed and saved on the MicroSD card by calling a function called save\_sd\_card(). The velocity is computed every defined interval, which is by default 10 milliseconds and then the graph function is called displaying the desired signal. After the last iteration of the for loop the TFT display is restarted.

On the lines 21 and 46 v\_max and F\_max are the variables to change the maximum range of the vertical axis of velocity in the when displaying the encoder signal and the force when displaying the load cell signal. Two Boolean variables called linear\_flag and rotational\_flag enable the linear or the rotational conversion of the encoder pulses.

The firmware developed for the device is available on:

<https://github.com/XaviCanoFerrer/Sport_Analyzer/tree/main/Firmware>

*Note: This is not meant to be an assembly instruction. Assembly instructions, detailed material lists, and construction files must be deposited in an appropriate repository (see Repositories document) and referenced in section “Build Details”.*

### Subsections

We encourage the use of subsections to highlight the modular functionality of the design or to highlight different design achievements. These questions might help you find appropriate subsections: What parts of the hardware could be used as a component of another project or a variation of this hardware? What solutions do you find most noteworthy given your experiences during the design iterations? What separate actions does the hardware perform?

# (2) Quality control

## Safety

Describe all relevant safety issues or reference to a risk assessment if included in the hardware documentation. Detail what safety considerations have been included in the hardware design and how these features have been tested, including any official safety standards or criteria that were used in this assessment. If appropriate, discuss the wider context of use of the hardware and safety issues or risks that may arise in the use environment.

## Calibration

If the hardware is used for measurements, please detail here how the reliability of measurements, or other hardware properties that are relevant for measurements, has been quantified and explain the results. Be clear about the processes or procedures used to compare the hardware to a standard, as well as the description of the standard calibrated against.

Detail the general procedures in place for users to calibrate their hardware before or during use. What methods can be used to relate user generated data to data from other sources?

*Note: Detailed instructions belong in documentation; here, provide insight into how and why the calibration is valid.*

### Subsections

We encourage the use of subsections within all sections to increase clarity.

## General testing

In this section, details can be provided on the testing of hardware functionalities, that are not directly essential for precision operation of the hardware in the given context (which are in turn, where applicable, handled under Calibration), such as automated movements to position the hardware, repeatability of tool exchanges, recyclability, water-tightness, weight or other possibly relevant characteristics. We encourage the authors to characterise all appropriate functionalities of the hardware, if not already described elsewhere (add reference instead). The testing should define the safe/reliable limits in which the components can be operated (e.g. step size and repeatability of linear motion, force ranges, ratio of devices with leaks when built in a workshop, etc). This will enhance the usability of the hardware or method in other contexts.

|  |  |  |
| --- | --- | --- |
|  | Value | Units |
| Sampling Frequency | 78 | Hz |
| Power consumption | 1.08 | W |
| Linear encoder resolution | 60 | µm |
| Load cell resolution | 0.1 | mN |
| Rotary encoder resolution | 2.6 | mrad |
| Maximum encoder measuring distance | 1.2 | m |
| Maximum load cell measuring force | 4.9 | kN |

*Again: Detailed instructions belong in documentation; here, provide a summary instead.*

# (3) Application

## Use case(s)

Describe at least one example of an application of your hardware. This should include some evidence of output, e.g. data produced by the use of the device or a picture of other types of results. Outline how the quality control in the previous section enables the use of the hardware in this context. We encourage the inclusion of experiment results or the reference to a publication (published or to-be-published) where these results are detailed. We also encourage pointers to ongoing work.

*Note: In the spirit of openness, we require authors to provide (or link to) datasets along with the submitted graphic representations. We do not impose arbitrary limits on inclusion of data so please include sufficient empirical detail and results to ensure your data can be easily verified, analysed and clearly interpreted by the wider scientific community.*

### Subsections

We encourage the demonstration of different use cases, divided by sub-sections to guide the reader.

## Reuse potential and adaptability

Please describe in as much detail as possible the ways in which the hardware could be reused by other researchers both within and outside of your field. This should include the use cases for the hardware, and also details of how the hardware might be modified or extended (including how contributors should contact you) if appropriate. Refer to section “Ease of build” where necessary.

Please provide your thoughts on the adaptability of the hardware design. What tools and procedures would it require for other users to modify your design without your help in order to adapt them for foreseeable or even unforeseeable use.

Also you must include details of what support mechanisms there are in place for this hardware and software (even if there is no support or support community).

# (4) Build Details

## Availability of materials and methods

Summarise what materials have been used to construct the hardware and what methods to process the materials as well as the assembly. Provide more details or references where important materials or methods are non-standard, not globally available or produced only by one manufacturer.

## Ease of build

Have any measures been taken in the design to make the hardware easy to build for other users e.g. reduction of parts, features in the design to make the hardware assembly more reliable?

## Operating software and peripherals

If hardware requires software, details on the operating software and programming language - Please include minimum version compatibility. Additional system requirements, e.g. memory, disk space, processor, input or output devices.

If the hardware does not require software, detail any required supporting processes or protocols required for use.

## Dependencies

E.g. other hardware or software projects, modular components, libraries, frameworks, incl. minimum version compatibility. Explicitly state if dependencies are proprietary / closed source.

## Hardware documentation and files location:

***Archive*** ***for hardware documentation and build files*** (required. See *Repositories* document for criteria, DocuBricks and alternative repositories.) *Note: We require the inclusion of modifiable design files as well as a detailed documentation of the design rational of the hardware with assembly instructions. This will be assessed as part of the journal peer review process.*

***Name:*** The name of the archive

***Persistent identifier:*** e.g. DOI, etc.

***Licence:*** Open hardware license under which the documentation and files are licensed - see *Essential author information* for more information

***Publisher:*** Name of the person who deposited the documentation

***Date published:*** dd/mm/yy

**Modifiable design files** (if different from above)

***Name:*** The name of the emulation environment

***Persistent identifier:*** e.g. DOI, handle, PURL, etc.

***Licence:*** Open license under which the software is licensed here

***Publisher:*** Name of the person who deposited the documentation

***Date published:*** dd/mm/yy

**Software code repository** (e.g. SourceForge, GitHub etc.) (required)

***Name:*** The name of the code repository

***Identifier:*** The identifier (or URI) used by the repository

***Licence:*** Open license under which the software is licensed

***Date published:*** dd/mm/yy

# (5) Discussion

## Conclusions

Conclusions, learned lessons from design iterations, learned lessons from use cases, summary of results.

## Future Work

Further work pursued by the authors or collaborators; known issues; suggestions for others to improve on the hardware design or testing,, given what you have learned from your design iterations.

## Paper author contributions

Task (e.g. design, assembly, use cases contribution, documentation, paper writing), contribution, author name.

## Acknowledgements

We thank Pau Gomez-Ferrer, Pier Sietti, Artur Lopez, Rainiero for the feedback of the functionality of the device.

Please add any relevant acknowledgements to anyone else who supported the project in which the hardware was created, but did not work directly on the hardware itself.

Please list anyone who helped to create the hardware and software (who may also not be an author of this paper), including their roles and affiliations.

## Funding statement

If the hardware resulted from funded research please give the funder and grant number.

## Competing interests

If any of the authors have any competing interests then these must be declared. The authors’ initials should be used to denote differing competing interests. For example: “BH has minority shares in [company name], which part funded the research grant for this project. All other authors have no competing interests." Or “BH is selling kits and parts connected to the here presented hardware via platform XX. A fundraising via Crowdfunding platform YY is planned to start commercialisation.”

If there are no competing interests, please add the statement:

“The authors declare that they have no competing interests.”

The authors declare that they have no competing interests.

## References

Please enter references in the Harvard style and include a DOI where available, citing them in the text with a number in square brackets, e.g.

[1] Bazuelo-Ruiz B, Padial P, García-Ramos A, Morales-Artacho AJ, Miranda MT, Feriche B. Predicting Maximal Dynamic Strength From the Load-Velocity Relationship in Squat Exercise. J Strength Cond Res. 2015 Jul;29(7):1999-2005. doi: 10.1519/JSC.0000000000000821. PMID: 25881572.

[2] Banyard, Harry G.; Nosaka, Kazunori; Haff, G. Gregory Reliability and Validity of the Load–Velocity Relationship to Predict the 1RM Back Squat, Journal of Strength and Conditioning Research: July 2017 - Volume 31 - Issue 7 - p 1897-1904 doi: 10.1519/JSC.0000000000001657

[3] Bosquet L, Porta-Benache J, Blais J. Validity of a Commercial Linear Encoder to Estimate Bench Press 1 RM from the Force-Velocity Relationship. J Sports Sci Med. 2010 Sep 1;9(3):459-63. PMID: 24149641; PMCID: PMC3761713.

[4] van den Tillaar R, Ball N. Validity and Reliability of Kinematics Measured with PUSH Band vs. Linear Encoder in Bench Press and Push-Ups. Sports (Basel). 2019 Sep 10;7(9):207. doi: 10.3390/sports7090207. PMID: 31509960; PMCID: PMC6784224.

[5] Ferro A, Floría P, Villacieros J, Muñoz-López A. Maximum velocity during loaded countermovement jumps obtained with an accelerometer, linear encoder and force platform: A comparison of technologies. J Biomech. 2019 Oct 11;95:109281. doi: 10.1016/j.jbiomech.2019.07.025. Epub 2019 Jul 31. PMID: 31471113.

[6] Illera-Domínguez V, Nuell S, Carmona G, Padullés JM, Padullés X, Lloret M, Cussó R, Alomar X, Cadefau JA. Early Functional and Morphological Muscle Adaptations During Short-Term Inertial-Squat Training. Front Physiol. 2018 Sep 10;9:1265. doi: 10.3389/fphys.2018.01265. PMID: 30246805; PMCID: PMC6139363.

[7] Lindemann U, Farahmand P, Klenk J, Blatzonis K, Becker C. Validity of linear encoder measurement of sit-to-stand performance power in older people. Physiotherapy. 2015 Sep;101(3):298-302. doi: 10.1016/j.physio.2014.12.005. Epub 2015 Mar 2. PMID: 25796540.

[8] Macaluso A, De Vito G. Comparison between young and older women in explosive power output and its determinants during a single leg-press action after optimisation of load. Eur J Appl Physiol. 2003 Nov;90(5-6):458-63. doi: 10.1007/s00421-003-0866-7. Epub 2003 Jul 19. PMID: 12883896.

[9] Fuentes AD, Smith CJ, Shoepe TC. Loading Patterns of Rubber-Based Resistance Bands across Distributors. Sports (Basel). 2019 Jan 16;7(1):21. doi: 10.3390/sports7010021. PMID: 30654430; PMCID: PMC6358948.

[10] Paul Stoffregen, “SPI library for Teensy & Arduino IDE” <https://github.com/PaulStoffregen/SPI>

[11] Paul Stoffregen, <https://github.com/PaulStoffregen/ILI9341_t3/blob/master/font_Arial.h>

[12] Paul Stoffregen, “Optimized ILI9341 TFT Library” <https://github.com/PaulStoffregen/ILI9341_t3>

[13] Thomas O Fredericks, “Debouncing library for Arduino and Wiring” <https://github.com/thomasfredericks/Bounce2>

[14] Paul Stoffregen, “Quadrature Encoder Library for Arduino” <https://github.com/PaulStoffregen/Encoder>

[15] Paul Stoffregen, <https://github.com/PaulStoffregen/SD>

[16] Bogdan Necula, “An Arduino library to interface the Avia Semiconductor HX711 24-Bit Analog-to-Digital Converter (ADC) for Weight Scales.” <https://github.com/bogde/HX711>

[17] Kris Kasprzak, “Functions to draw graphs on LCD” <https://github.com/KrisKasprzak/GraphingFunction/blob/master/Graph.ino>

[18] Jasper van Loenen, “” <http://javl.github.io/image2cpp/>

[] Bosco C, Luhtanen P, Komi PV. A simple method for measurement of mechanical power in jumping. Eur J Appl Physiol Occup Physiol. 1983;50(2):273-82. doi: 10.1007/BF00422166. PMID: 6681758.

Martin, L 2010 Bombs, bodies and biopolitics: Securitizing the subject at airport security. Social and Cultural Geography, 11(1): 17-34. DOI: http://dx.doi.org/10.1080/14649360903414585

NOTE: Please include DOIs for all journal articles where possible.

# Copyright notice

Authors who publish with this journal agree to the following terms:

Authors retain copyright and grant the journal right of first publication with the work simultaneously licensed under a [Creative Commons Attribution License](http://creativecommons.org/licenses/by/3.0/) that allows others to share the work with an acknowledgement of the work's authorship and initial publication in this journal.

Authors are able to enter into separate, additional contractual arrangements for the non-exclusive distribution of the journal's published version of the work (e.g., post it to an institutional repository or publish it in a book), with an acknowledgement of its initial publication in this journal.

By submitting this paper you agree to the terms of this Copyright Notice, which will apply to this submission if and when it is published by this journal.